

**AMENDMENTS TO THE CLAIMS:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**LISTING OF CLAIMS:**

1. (Currently Amended) A method of managing an original executable code forming a program to be downloaded into a reprogrammable on-board computer system in a microprocessor card, said code possessing a cryptographic signature and being executable by the microprocessor of the on-board system after verification by the latter of the validity of said signature, said method comprising the following steps:

- off card:

- identifying a modified executable code that is adapted to a predefined specific use, and corresponds to the original code, ~~adapted to a predefined specific use, and~~

~~-from variations between the data of the original code and the corresponding modified code,~~ calculating a software component according to the original intermediate code and the corresponding modified intermediate code identified for the predefined specific use, which, the software component, when it is applied to the original code, ~~makes making~~ it possible to ~~reconstruct produce~~ the modified code;

- signing said software component;

- downloading the signed original code and the signed software component into the card; and

- on card:

- verifying the signatures respectively of the original code and of the software component, and

- applying the software component to the original code so as to ~~reconstruct produce~~ the modified intermediate code for its execution by the microprocessor.

2. (Previously Presented) A method according to claim 1, wherein the original executable code consists of an intermediate code, executable by the on-board system microprocessor by means of a virtual machine for interpreting this intermediate code.

3. (Currently Amended) A method according to claim 2, wherein the virtual machine is provided with an execution stack and wherein the downloaded software component, which is applied on card to the original intermediate code, makes it possible to ~~reconstruct~~ produce a modified intermediate code a priori satisfying the verification criteria for said intermediate code according to which the operands of each instruction of said code belong to the data types manipulated by this instruction and, on each target switching instruction, the execution stack of the virtual machine is empty.

4. (Currently Amended) A method according to claim 3, wherein the modified intermediate code obtained by the application of the software component is verified, before ~~its~~ the execution by the microprocessor by means of the virtual machine, according to a process verifying that the modified intermediate code satisfies the verification criteria.

5. (Currently Amended) A method according to claim 1, wherein the downloaded software component, applied on card to the original code, makes it possible to ~~reconstruct~~ produce a modified code so that ~~its~~ execution of the modified code is more rapid compared with that of the original code.

6. (Currently Amended) A method according to claim 1, wherein the downloaded software component, applied on card to the original code, makes it possible to ~~reconstruct~~ produce a modified code so that it procures an optimization in terms of size compared with the original code.